BIOINFORMATICS

Vol. 00 no. 00 2003
Pages 1–7

Recovering haplotype structure through recombination and gene conversion

Mathieu Lajoie [a], Nadia El-Mabrouk [b]

[a]Département d'informatique et de recherche opérationnelle, Université de Montréal, CP 6128 Succursale Centre-ville, Montréal, QC, H3C 3J7, Canada, [b]DIRO, Université de Montréal, CP 6128 Succursale Centre-ville, Montréal, QC, H3C 3J7, Canada.

ABSTRACT

Motivation Understanding haplotype and genotype evolution subject to mutation, recombination and gene conversion is fundamental to understand genetic specificities of human populations and hereditary bases of complex disorders. The goal of this project is to develop new algorithmic tools assisting the reconstruction of historical relationships between haplotypes, and the inference of haplotypes from genotypes.

Results We present two algorithms. The first one finds an optimal pathway of mutations, recombinations and gene conversions leading to a given **haplotype** of size m from a set of h putative ancestral haplotypes. It runs in time $O(mhs^2)$, where s is the maximum number of contiguous sites that can be exchanged in a single gene conversion. The second one finds an optimal pathway of mutations and recombinations leading to a given **genotype**, and runs in time $O(mh^2)$. Both algorithms are based on a penalty score model and use a dynamic programming approach. We apply the second one to the problem of inferring haplotypes from genotypes, and show how it can be useful to resolve "hard" genotypes, when the underlying pair of haplotypes differ substantially from frequent ones.

Availability The algorithms have been implemented in JAVA, and are available on request.

Contact: mabrouk@iro.umontreal.ca

1 INTRODUCTION

Since the sequencing of the human genome, a great effort has been deployed to characterize allelic diversity at the nucleotide level, represented by single nucleotide polymorphisms (SNPs). Having access to these genetic markers is fundamental for epidemiological studies in the quest of hereditary bases of complex disorders. However, it is less the individual variants that counts than their overall organization along the chromosomes. A haplotype is a string of polymorphic sites along a DNA sequence (Figure 1). Preliminary to any human genetic project, is the acquirement of a haplotype dataset. However, in diploid organisms, it is not feasible to examine the two copies of a chromosome separately. Rather, it is the (less informative) *genotype*, e.g. the combination of the two complementary chromosomes, that is obtained. The haplotyping problem is then to extract, from this information, individual haplotypes.

In addition to characterizing allelic diversity created by spontaneous mutations, understanding how individual variants are redistributed accross populations and organized in blocks has been shown fundamental in the study of human diversity and disease inference [34, 8, 7]. Recombination redistribute individual variants among copies the homologous chromosomes [9, 20], and gene conversion occur when, during crossing-over, the Holliday junction returns to the initial configuration rather than being resolved such that chromatids cross and thus accomplish the recombination (Figure 2). Gene-conversion can be seen as two either concomitant or successive recombinations. However, at a short distance, a double crossing-over within a single meiosis is sterically impossible, and it is gene-conversion that can be invoked to explain the data [27, 15, 1, 21]. To understand the genealogical relationships between haplotypes and their "blocky" structures, it is thus important to study their process of evolution subject to mutation, recombination and gene conversion.

Prior work on recombination and gene conversion has largely focused on statistical tests estimating the recombination events [13, 18, 33], and on reconstructing the coalescent with recombination and/or gene conversion, based on statistical models assuming constant population length, random mating, and given mutation and rearrangement rates per generation [10, 30, 29, 31]. Other methods based on algorithmic optimization have been considered for the reconstruction of a plausible genealogy of haplotypes [16, 28, 25, 22, 32], but most of these reconstruction problems have been shown NP-hard. Consequently, simplified evolutionary models have been considered. In particular, because of a relatively simple pattern of haplotype diversity in the human genome with a domination of few common haplotypes [14, 17, 19, 26], the complexity of the haplotype network can be reduced by considering the most frequent haplotypes as the most likely to recombine.

In this paper, we address the problem of inferring the most realistic pathway of mutations, recombinations and gene conversions generating a given haplotype from a set of h putative

ancestral haplotypes of size m. Previous dynamic programming methods have been developed in the absence of gene conversion [4, 22]. In [3], we formalized the problem and described the whole set of pathways involving a minimum number of recombinations and gene conversions leading to a haplotype, and described a partial method to find a representative pathway. Here, we consider the more general case involving a penalty score model for mutations, recombinations and gene conversions, and describe an optimized dynamic programming algorithm that runs in time $O(mhs^2)$, where s is the maximum size of a gene conversion. This algorithm is described in Section 2.

In the second part of this paper, we present a new algorithm based on the same evolutionary model, to infer haplotypes from genotypes. Several approaches have been developed for this purpose, beginning with the Clark's parsimony approach [2] and maximum likelihood approaches [6]. In the absence of recombinations, more combinatorial approaches based on the perfect phylogeny model have been developed [11, 5]. In the general case, the most widely used approach is PHASE, based on a Gibbs sampling method [24, 23]. In most cases, the software reports a set of accurate haplotype pairs. However some genotypes give rise to ambiguous results, e.g. many possible haplotype pairs with low probabilities. Moreover time before convergence may be very long. In section 3, we present an efficient method, which runs in time (mh^2), to resolve a given genotype with respect to a set of known haplotypes. In Section 4, we give some preliminary results demonstrating the accuracy of this method for genotypes that have been revealed problematic for PHASE.

2 RECOVERING RECOMBINATION AND GENE CONVERSION PATHWAYS

We describe an algorithm that finds an optimal (least score) pathway of mutations, recombinations and gene conversions generating a given haplotype from a set HAP of known haplotypes. This algorithm can be seen as a generalization of the one described in [22] including gene conversions.

The classical methods for inferring historical relationships between haplotypes assume an infinite site mutation model: at each SNP site, a mutation only happened once in human history. In other words, recurrent mutations and back mutations are forbidden. Here, we consider a relaxed model which allows for recurrent and back mutations.

2.1 The model and notations

A **haplotype** of size m is a string of symbols which models m single nucleotide polymorphisms (SNPs) on a chromosomal segment. SNPs are usually bi-allelic such that in a population, only two nucleotides are observed at each site. Therefore, haplotypes can be represented as binary strings of 0's (ancestral alleles) and 1's (new alleles) (Figure 1).

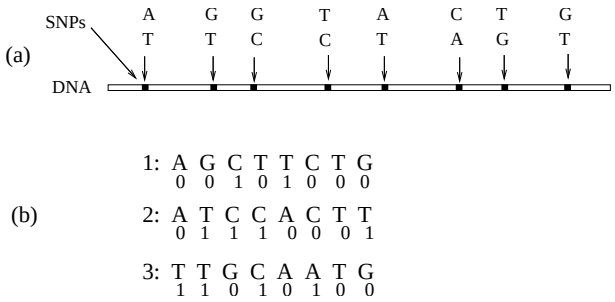

Fig. 1. (a) A genomic sequence with its polymorphic sites indicated by bold squares; (b) Three possible haplotypes found in the population, with their representations as binary strings, assuming that upper alleles represent ancestral ones.

A **recombination** between two haplotypes H_1 and H_2 can be modeled as an operation that breaks up H_1 and H_2 between sites i and $i-1$, and exchanges the two terminal parts of H_1 and H_2. (Figure 2).

A **gene conversion** between H_1 and H_2 is an operation that breaks up H_1 and H_2 in three parts each by choosing the same two pairs of adjacent sites in the two haplotypes, $i, i-1$ and $j, j-1$, and exchanges the two middle parts of H_1 and H_2 (Figure 2).

As only one of the resulting haplotypes is transmitted, a recombination or a gene conversion can be represented as $H_1, H_2 \longrightarrow H_3$, where H_1, H_2, H_3 are three haplotypes.

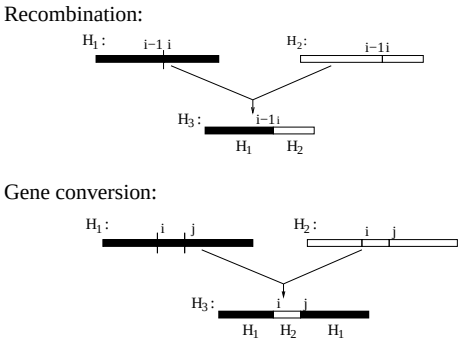

Fig. 2. The recombination and gene conversion mechanisms

Each SNP represents a mutation that has affected one haplotype in the population. Therefore, if recurrent mutations are ignored, then allelic changes can be explained solely by recombinations and gene conversions. In this paper, recurrent mutations are allowed, and we call a **mutation** an event that changes a 0 into a 1 or a 1 into a 0 in a haplotype.

In [22], Schwartz *et. al.* have considered a simplified probabilistic model allowing to evaluate a recombination and mutation pathway leading to a given haplotype. However, assigning the appropriate probabilities is an open problem by

itself. In this paper, we consider an alternative approach, by attributing penalty scores for mutations, recombinations and gene conversions.

The penalty score model is based on the following inputs:

1. MUT is the score of a (recurrent) mutation at any site in any haplotype.
2. REC(i) specifies the score of a recombination between sites $i-1$ and i. This value can be evaluated from the nucleotide distance separating these sites.
3. GC(i,j) is the score of a gene conversion between the two sites i and j. This value depends on the number of nucleotides separating sites $i+1$ and $j-1$. We also define the parameter s representing the maximum site length of a gene conversion, $l=(j-i)-1$, that is the maximum number of sites that can be exchange in a single gene conversion. This value, which depends on the nucleotide distances between the sites in the considered haplotypes, is usually small and serves as a bound for an efficient algorithmic complexity.
4. FREQ(H_p) is the score for choosing a particular haplotype H_p as part of the solution. We use the negative log-frequency of H_p.

2.2 The algorithm

To simplify the ensuing algorithmic developments, we recode the haplotypes in a way allowing to reformulate the problem as one of generating the **unitary haplotype**, that is the haplotype H such that $H[i]=1$ for any $1\leq i\leq m$. Let HAP be the set of h haplotypes of size m (Figure 3).

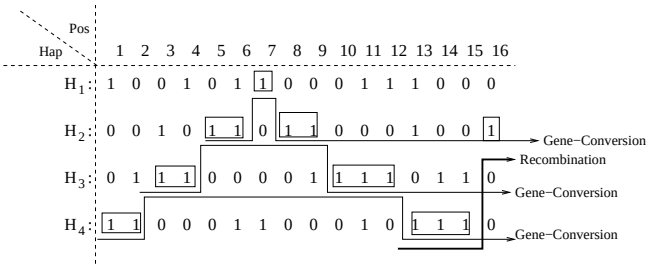

Fig. 3. A possible pathway generating the unitary haplotype from the set HAP = $\{H_1, H_2, H_3, H_4\}$, with three gene conversions and one recombination.

We denote $H_p[i..j]=H_p[i]\cdots H_p[j]$, for $1\leq i\leq j\leq m$. In other words, $H_p[i..j]$ is the subsequence of the haplotype H_p of HAP beginning at position i and ending at position j.

We denote by HAP$[i..j]$ the set $\{H_p[i..j]$, for $1\leq p\leq h\}$.

A pathway generating $H[i..j]$ is said to **end at haplotype** H_p if the last suffix of $H[i..j]$ comes from H_p.

To compute the minimal penalty score C of a pathway generating H from HAP, we recursively compute the scores $C(1,j)$ of the optimal pathways giving rise to the unitary haplotypes $H[1..j]$ from the set HAP$[1..j]$, for $1\leq j\leq m$.

Let $C_p(i,j)$ be the score of an optimal pathway R giving rise to $H[i..j]$ and ending at haplotype H_p. Then

$$C(i,j)=\min\{C_p(i,j),\ \text{for } 1\leq p\leq h\}$$

We show how to compute $C_p(i,j)$ for $i<j$. The case $i>j$ is symmetrical and obtained in the same way, but considering reverse haplotypes (red from right to left).

Suppose first that $H_p[j]=1$. Then $C_p(i,j)$ is one of the following:

1. $C_p(i,j)=C_p(i,j-1)$: just extend the haplotype H_p one position right.
2. If the last event of R is a recombination at position j, then $C_p(i,j)=C(i,j-1)+\text{REC}(j)+\text{FREQ}(H_p)$.
3. If the last event of R is a gene conversion between positions k and j, then $C_p(i,j)=C_p(i,k)+\text{CG}(k,j)+C(k+1,j-1)$. This case can happen only for $j>2$.

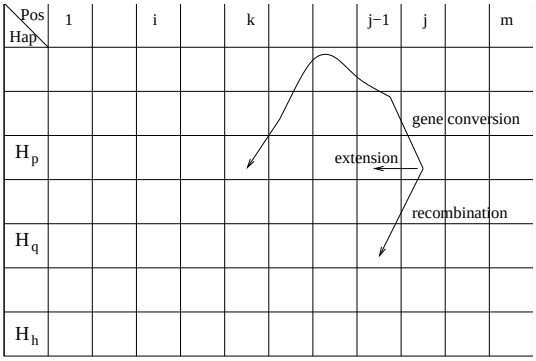

Fig. 4. The main dynamic programming table and three possible cases for the last event of a path giving rise to $H[1..j]$, with score $C_p(1,j)$.

If $H_p[j]=0$, an additional mutational event is necessary to transform $H_p[j]$ to 1.

Therefore, if we denote: $M_p(j)=\begin{cases}0\ if\ H_p[j]=1\\ \text{MUT otherwise}\end{cases}$

$$\begin{aligned}C_p(i,j)=\quad & M_p(j)+\min\{C_p(i,j-1),\\ & C(i,j-1)+\text{REC}(j)+\text{FREQ}(H_p),\\ & \min_{j-1-s\leq k\leq j-2}\{C_p(i,k)+\text{CG}(k,j)+C(k+1,j-1)\}\}\end{aligned}$$

The basic cases are $C_p(i,i)=\text{FREQ}(p)+M_p(i)$ for $1\leq i\leq m$ and the final pathway is the one leading to the score $C=C(1,m)$. The resulting algorithm is described in Figure 5.

Complexity: For each column j of the main dynamic programming table, $1\leq j\leq m$, the algorithm is subdivided into two parts:

Initialization:
For $i = 1$ **to** m **do**
  For $j = 1$ **to** m **do**
    $C(i,j) = \infty$;
  For $p = 1$ **to** h **do**
    $C_p(i,i) = \text{FREQ}(p) + M_p(i)$;
    $C(i,i) = \min(C(i,i), C_p(i,i))$;
For each column of the main dynamic programming table:
For $j = 2$ **to** m **do**
  For each line:
  For $p = 1$ **to** h **do**
    $C_p(1,j) = \min(C(1,j-1) + \text{REC}(j), C_p(1,j-1))$;
    For each of the s columns preceding column j :
    For $k = j-2$ **down-to** $j-1-s$ **do if** $k > 0$
      $\text{CG} = C_p(1,k) + \text{CG}(k,j) + C(j-1,k+1)$;
      $C_p(1,j) = \min(C_p(1,j), \text{CG})$;
    End For (k)
    $C_p(1,j) = C_p(1,j) + M_p(j)$;
    $C(1,j) = \min(C(1,j), C_p(1,j))$;
  End For (p)

  Consider "reverse" reconstruction, beginning at position j and a different table for storing the $C_*(j,*)$ values;
  For $j' = j-1$ **down to** $j-s+1$ **do**
    For $p = 1$ **to** h **do**
    $C_p(j,j') = \min(C(1,j-1) + \text{REC}(j), C_p(1,j-1))$;
      For $k = j'+2$ **to** j **do if** $k \leq j$
        $\text{CG} = C_p(j,k) + \text{CG}(j,k) + C(j'+1,k-1)$
        $C_p(j,j') = \min(C_p(j,j'), CG)$;
      End For (k)
      $C_p(j,j') = C_p(j,j') + M_p(j')$;
      $C(j,j') = \min(C(j,j'), C_p(j,j'))$;
    End For (p)
  End For (j')
End For (j)

Fig. 5. Dynamic algorithm for the computation of $C(i,j)$, $1 \leq i,j \leq m$. The value of the optimal path leading to H is given by $C(1,m)$.

- The computation of $C(1,j)$, that is $\min_{1 \leq p \leq h} C_p(1,j)$. For each haplotype p, $1 \leq p \leq h$, computing the value of $C_p(1,j)$ requires to consider the s values $C_p(j-1,k+1)$, for $j-1-s \leq k \leq j-2$. Therefore, the complexity of this part is $O(hs)$.
- The computation of $C(j,j')$, for the s columns j' preceding j. As for the previous step, for each haplotype p, $1 \leq p \leq h$, the computation of $C_p(j,j')$ requires to consider all the values $C(j'+1,k-1)$, for $j'+2 \leq k \leq j$. Therefore, the complexity of this part is $O(hs^2)$.

The total complexity of the algorithm is thus $O(m(hs + hs^2)) = O(mhs^2)$.

3 RECONSTRUCTING HAPLOTYPES FROM GENOTYPES

A genotype is commonly represented as a sequence of 0, 1 and 2, where 0 and 1 correspond to homozygous sites (both haplotypes have the same allele, i.e. two 0s or two 1s), and 2 represents heterozygous sites (a 0 on one haplotype and a 1 on the other). The haplotyping problem is to phase the heterozygous sites, that is to determine on which of the two haplotypes is the 0 allele and the 1 allele (Figure 6).

Genotype:	2	1	2	0	1	2	0	1	2	
Resolution:	0	1	1	0	1	1	0	1	1	←H1
	1	1	0	0	1	0	0	1	0	←H2

Fig. 6. A genotype G and two haplotypes representing a possible resolution of G.

The most accurate haplotyping methods follow (at least implicitly) these principles:

1. If an unresolved genotype can be explained by a pair of already known haplotypes, then this pair is likely to be the right one. In case of many possible pairs, the most likely one depends on the frequencies of the haplotypes in the population.
2. Otherwise, at least one new haplotype is inferred. Any new haplotype should be as close as possible, with respect to the genetic model, to the other ones in the population.

In particular, PHASE uses a Gibbs sampling method, beginning with an arbitrary resolution of the set of genotypes, and successively updating each pair of haplotypes with respect to the set of all other haplotypes. The whole process is repeated for a fixed number of times, or until convergence. Pairs of haplotypes are then reported with their associated probabilities. However, in some cases convergence is not reached, and some genotypes give rise to many possible haplotype pairs with low probabilities. In this case, alternative methods allowing to solve ambiguous genotypes may be valuable.

Here, we present a new method to resolve a single genotype in light of a set of known (or inferred) haplotypes. The first step is to find an optimal pathway of mutations and recombinations leading from the known haplotypes to the target genotype. This pathway in then used to infer the haplotype pair.

The penalty model is based on the same three inputs MUT, $\text{REC}(i)$ and $\text{FREQ}(H_p)$ defined in the preceding section.

3.1 Finding an optimal pathway

We generate the set G of all possible genotypes that can be obtained from two haplotypes of HAP. More precisely, $\mathsf{G} = \{G_{p,q} = (H_p, H_q), \text{ for } 1 \leq p \leq q \leq h\}$. The problem is then to find the recombination and mutation pathway of minimal score C generating the unresolved genotype G from G. For $1 \leq j \leq m$, let $C(j)$ be the score of an optimal pathway giving rise to $G[1..j]$ from the set $\mathsf{G}[1..j]$, and $C_{p,q}(j)$ the score of such a path ending at genotype $G_{p,q}$. Then

$$C(j) = \min\{C_{p,q}(j), \text{ for } 1 \leq p,q \leq h\}$$

Let R be an optimal pathway generating $G[1..j]$ with score $C_{p,q}(j)$. Suppose first that $G_{p,q}[j] = G[j]$. Then $C_{p,q}(j)$ is computed from some $C_{p',q'}(j-1)$ as follows:

1. If $p = p'$ and $q = q'$ (or similarly $p = q'$ and $q = p'$), then we just extend the genotype $G_{p,q}$ one position right. Thus, $C_{p,q}(j) = C_{p,q}(j-1)$.
2. Otherwise, if $p = q$ and $p' = q'$, then there is one recombination between H_p and $H_{p'}$ (or similarly between H_q and $H_{q'}$), and $C_{p,p}(j) = C_{p',p'}(j-1) + \text{REC}(j) + \text{FREQ}(p)$.
3. Otherwise, if $\{p,q\} \cap \{p',q'\} = \emptyset$, then two recombinations at site j are necessary, and $C_{p,q}(j) = C_{p',q'}(j-1) + 2.\text{REC}(j) + \text{FREQ}(p) + \text{FREQ}(q)$.
4. Otherwise, $|\{p,q\} \cap \{p',q'\}| = 1$. W.l.o.g., assume $p = p'$. Then there is a recombination between H_q and $H_{q'}$, and $C_{p,q}(j) = C_{p',q'}(j-1) + \text{REC}(j) + \text{FREQ}(q)$.

Let $C'_{p',q'}(j)$ be the value obtained from the preceding formula. If $G_{p,q}[j] \neq G[j]$, then mutation penalties should be added as follows:

a. If the values of $G_{p,q}[j]$ and $G[j]$ are in $\{0,1\}$ and $p \neq q$, then two mutations are necessary and $C_{p',q'}(j) = C'_{p',q'}(j) + 2.\text{MUT}$.
b. If the values of $G_{p,q}[j]$ and $G[j]$ are in $\{0,1\}$, but $p = q$, then only one mutation is necessary and $C_{p',q'}(j) = C'_{p',q'}(j) + \text{MUT}$.
c. If $G_{p,q}(j)$ or $G(j)$ has value 2, then just one mutation is required, and $C(j) = C'(j) + \text{MUT}$.

The final result is $C = C(m)$ with the associate path.

Complexity: It is possible to compute each value $C_{p,q}(j)$ in constant time, since REC(j), MUT, FREQ(p) and FREQ(q) do not depend on p', neither on q'. All we need is to compute (at no additional cost) the following values, which correspond to the best choices of genotypes for the three possible scenarios of recombination:

- $\min_{p',q'}(C_{p',q'}(j-1))$
- $\min_{p'}(C_{p',q}(j-1))$
- $\min_{q'}(C_{p,q'}(j-1))$

Since $1 \leq p \leq q \leq h$ and $1 \leq j \leq m$, the global complexity of the algorithm is in $O(mh^2)$.

3.2 Inferring haplotype pairs

In the case of a single recombination at one site (cases 2 and 4 above), there is no ambiguity to deduce the corresponding haplotype pair. For example, suppose we have a genotype $G = 0221$, the haplotypes $H_1 = 1111$, $H_2 = 0000$, $H_3 = 0101$ and the following optimal path:

$$\mathsf{R} = \frac{H_3}{H_2}\frac{H_3}{H_2}\frac{H_3}{H_1}\frac{H_3}{H_1}$$

In this case, inferring the underlying pair of haplotypes is straightforward:

$$G = \frac{0101}{0011}$$

However, in the case of two recombinations at the same site (case 3 above), the phase can not be deduced. For example:

$$\frac{H_4}{H_3}\frac{H_4}{H_3}\frac{H_1}{H_2}\frac{H_1}{H_2} \equiv \frac{H_4}{H_3}\frac{H_4}{H_3}\frac{H_2}{H_1}\frac{H_2}{H_1}$$

In this case, additional information should be considered to choose between the two different scenarios. Additional penalties can also be added to favor informative pathways.

The situation with mutations is similar. Cases (a) and (b) leave no ambiguities, where as case (c) do not allow to decide on which of the two haplotype the mutation should be placed. Here also, it is possible to prevent this case by adding an extra penalty to this scenario. If ambiguous mutations persist, we chose to place them on the new haplotype that is the farthest one from known haplotypes.

4 EXPERIMENTS

We simulated various independent data sets under the infinite-sites model by using the Hudson's program [12]. Each set consisted of 50 genotypes obtained by random pairing of 100 haplotypes, assuming a panmictic constant size population. For each set, we used PHASE version 2.1 with default parameters. The software returns the best possible pairs of haplotypes explaining each genotype, with a probability associated to each pair. We considered a genotype as *ambiguous* when all its best haplotype pairs were reported with probabilities of 0.3 or less. For other genotypes, we stored all pairs of haplotypes reported with probabilities ≥ 0.3 in the set HAP of known haplotypes. We finally applied our method to the ambiguous genotypes. We then compared the predicted pairs with the true ones, and reported the number of correctly resolved genotypes for each method. All tests were done with penalty 11 for mutations and 10 for recombinations.

Table 1 shows the results obtained on datasets generated with different recombination parameters. In each case, the number of ambiguous genotypes correctly resolved by our algorithm is higher. However, the impact on the overall performance remains small. Moreover, these preliminary results do not allow to evaluate the effect of recombination rates on the accuracy of our method.

We then performed similar tests on longer haplotypes (Table 4). In this case, the number of ambiguous genotypes correctly resolved by our algorithm is significantly higher. Moreover, solving each ambiguous haplotypes required no more than few seconds.

	Ambiguous genotypes		
		Correctly resolved	
$4N_e r$	Total	by PHASE	by ours
16	49	12	24
24	48	20	21
32	55	15	23
40	54	16	18

Table 1. Results summed over 30 independent datasets for different values of the recombination parameter $R = 4N_e r$ (120 independent data sets in total). We fixed the mutation parameter to $\theta = 4N_e\mu = 16$. The size of the resulting haplotypes varies from 60 to 100 polymorphic sites.

	Ambiguous genotypes		
		Correctly resolved	
Dataset	Total	by PHASE	by ours
1	3	0	2
2	2	1	2
3	3	0	1
4	3	0	2
5	7	1	7
6	3	0	0
7	4	1	1
8	3	0	1
9	4	1	2
10	7	4	5
11	6	1	4
12	5	1	2
13	5	1	1
14	4	0	3
15	2	0	0
16	3	1	3
17	4	0	1
18	6	2	4
19	3	0	1
20	7	0	2
Total	84	14	44

Table 2. Results obtained for 20 datasets generated with the parameters $4N_e\mu = 4N_e r = 32$. The size of the resulting haplotypes varies from 125 to 185 polymorphic sites.

5 CONCLUSION

We have developed formal tools to find probable evolutionary pathways giving rise to a given haplotype or genotype, under a realistic model involving mutations, recombinations and gene conversions. This is the first step toward a more general heuristic allowing to reconstruct the complete evolutionary network connecting all haplotypes. Another important application would be to estimate the frequencies of recombinations compared to those of gene conversions of different types, based on population data.

A direct application to the haplotyping problem has been presented. The time efficiency of the developed algorithm makes it useful to solve "hard" genotypes that give rise to ambiguous results with statistical methods. The preliminary results obtained are encouraging and reveal a good performance for long genotypes. However more experiments have to be performed with different penalty scores, to test the method.

At this stage, gene conversions were not included in our evolutionary model for haplotyping, as our method do not naturally extend to that case. However, this should have a limited effect on the final solution, as gene conversions usually involve one or two polymorphic sites, and thus can be treated as mutations.

REFERENCES

[1]P. Andolfatto and M. Nordborg. The effect of gene conversion on intralocus associations. *Genetics*, 148:1397- 1399, 1998.

[2]A. Clark. Inference of haplotypes from PCR-amplified samples of diploid populations. *Mol. Biol. Evol.*, 7:111- 122, 1990.

[3]N. El-Mabrouk. Deriving haplotypes through recombination and gene conversion. *Journal of computational Biology*, 2(2):241-256, 2004.

[4]N. El-Mabrouk and D. Labuda. Haplotypes histories as pathways of recombinations. *Bioinformatics*, 20:1836-1841, 2004.

[5]E. Eskin, E. Halperin, and K. Karp. Large scale reconstruction of haplotypes from genotype data. In *Proceedings of the seventh annual international conference on reserach in Computational molecular biology (RECOMB)*. ACM Press, 2003.

[6]L. Excoffier and M. Slatkin. Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. *Mol. Biol. Evol.*, 12(5):921-927, 1995.

[7]S.B. Gabriel, S.F. Schaffner, H. Nguyen H, J.M. Moore, J. Roy, B. Blumenstiel, J. Higgins, M. DeFelice, A. Lochner A, M. Faggart, S.N. Liu-Cordero, C. Rotimi, A. Adeyemo, R. Cooper, R. Ward, E.S. Lander, M.J. Daly, and D. Altshuler. The structure of haplotype blocks in the human genome. *Science*, 296(5576):2225 - 2229, 2002.

[8]G. Greenspan and D. Geiger. Model-based inference of haplotype block variation. In W. Miller, M. Vingron, and S. Istrail, editors, *Proceedings of the seventh annual international conference on reserach in Computational molecular biology (RECOMB)*, pages 131 - 137. ACM Press, 2003.

[9]T.A. Greenwood, B.K.Rana, and N.J. Schork. Human haplotype block sizes are negatively correlated with recombination rates. *Genome Research*, 14:1358-1361, 2004.

[10]R.C. Griffiths and P. Marjoram. Ancestral inference from samples of DNA sequences with recombination. *Jounal of Computational Biology*, 3(4):479-502, 1996.

[11]D. Gusfield. Haplotyping as perfect phylogeny: Conceptual framework and efficient solutions. In *Proceedings of the sixth annual international conference on reserach in Computational molecular biology (RECOMB)*, pages 166 - 175. ACM Press,

2002.

[12]R.R. Hudson. Generating samples under a wright-fisher neutral model of genetic variation. *Bioinformatics*, 18:337-338, 2002.

[13]R.R. Hudson and N.L. Kaplan. Statistical properties of the number of recombination events in the history of a sample of DNA sequences. *Genetics*, 111:147-164, 1985.

[14]J. Jaruzelska, E. Zietkiewicz, M. Batzer, D.E. Cole, J.P. Moisan, R. Scozzari, S. Tavaré, and D. Labuda. Spatial and temporal distribution of the neutral polymorphisms in the last ZFX intron: analysis of the haplotype structure and genealogy. *Genetics*, 152:1091-101, 1999.

[15]A.J. Jeffreys and C.A. May. Intense and highly localized gene conversion activity in human meiotic crossover hot spots. *Nat Genet.*, 36(2):151- 156, 2004.

[16]J. Kececioglu and D. Gusfield. Reconstructing a history of recombinations from a set of sequences. *Discrete Applied Mathematics*, 88:239-260, 1998.

[17]D. Labuda, E. Zietkiewicz, and V. Yotova. Archaic lineages in the history of modern humans. *Genetics*, 156:799- 808, 2000.

[18]S.R. Myers and R.C. Griffiths. Bounds on the minimum number of recombination events in a sampke history. *Genetics*, 2002.

[19]M.V. Osier, A.J. Pakstis, H. Soodyall, D. Comas, D. Goldman, A. Odunsi, F. Okonofua, J. Parnas, L.O. Schulz, J. Bertranpetit, B. Bonne-Tamir, R.B. Lu, J.R. Kidd, and K.K. Kidd. A global perspective on genetic variation at the ADH genes reveals unusual patterns of linkage disequilibrium and diversity. *Am. J. Hum. Genet.*, 71:84- 99, 2002.

[20]D. Posada, K.A. Crandall, and E.C. Holmes. Recombination in evolutionary genomics. *Annu. Rev. Genet.*, 36:75 - 97, 2002.

[21]M. Przeworski and J.D. Wall. Why is there so little intragenic linkage disequilibrium in humans? *Genet. Res., Camb.*, 77:143-151, 2001.

[22]R. Schwartz, A.G. Clark, and S. Istrail. Methods for inferring block-wise ancestral history from haploid sequences - The haplotype coloring problem. In R. Guigó and D. Gusfield, editors, *Second International Workshop, Algorithms in Bioinformatics (WABI'02)*, volume 2452 of *LNCS*, pages 44-59. Springer, 2002.

[23]M. Stephens and P. Donnelly. A comparison of bayesian methods for haplotype reconstruction from population genotype data. *Am. J. Hum. Genet.*, 73(4):1162- 1169, 2003.

[24]M. Stephens, N.J. Smith, and P. Donnelly. A new statistical method for haplotype reconstruction from population data. *Am. J. Hum. Genet.*, 68(4):978- 989, 2001.

[25]E. Ukkonen. Finding founder sequences from a set of recombinants. In R. Guigó and D. Gusfield, editors, *Second International Workshop, Algorithms in Bioinformatics (WABI'02)*, volume 2452 of *LNCS*, pages 277-286. Springer, 2002.

[26]B.C. Verrelli, J.H. McDonald, G. Argyropoulos, G. Destro-Bisol, A. Froment, A. Drousiotou, G. Lefranc, A.N. Helal, J. Loiselet, and S.A. Tishkoff. Evidence for balancing selection from nucleotide sequence analyses of human G6PD. *Am. J. Hum. Genet.*, 71:1112–28, 2002.

[27]J.D. Wall. Close look at gene conversion hot spots. *Nature Genetics*, 36(2):114 - 115, 2004.

[28]L. Wang, K. Zhang, and L. Zhang. Perfect phylogenetic networks with recombination. *Journal of Computational Biology*, 8(1):69-78, 2001.

[29]C. Wiuf and J. Hein. The ancestry of a sample of sequences subject to recombination. *Genetics*, 151:1217-1228, 1999.

[30]C. Wiuf and J. Hein. Recombination as a point process along sequences. *Theoretical Population Biology*, 55:248-259, 1999.

[31]C. Wiuf and J. Hein. The coalescent with gene conversion. *Genetics*, 155:451-462, 2000.

[32]S. Wu and X. Gu. A greedy algorithm for optimal recombination. In J. Wang, editor, *COCOON*, volume 2001 of *LNCS*, pages 87-90. Springer-Verlag, 2001.

[33]Y.S.Song and J.Hein. On the minimum number of recombination events in the evolutionary history of dna sequences. *J. Math. Biol.*, 48:160- 186, 2004.

[34]K. Zhang, F. Sun, M.S. Waterman, and T. Chen. Dynamic programming algorithms for haplotype block partitioning: applications to human chromosome 21 haplotype data. In W. Miller, M. Vingron, and S. Istrail, editors, *Proceedings of the seventh annual international conference on reserach in Computational molecular biology (RECOMB)*, pages 332 - 340. ACM Press, 2003.